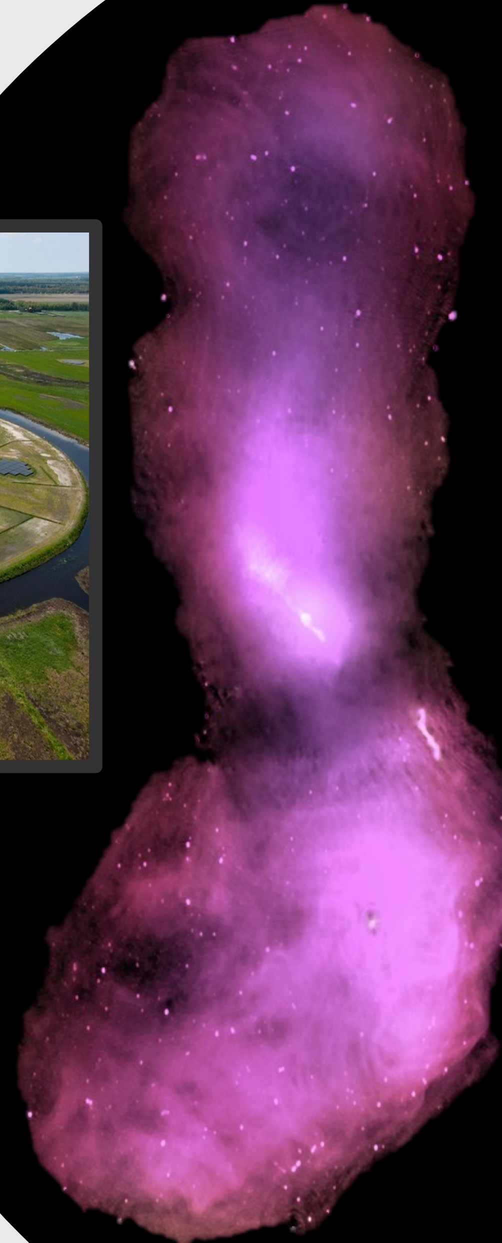
ASTRON

Introduction to Imaging



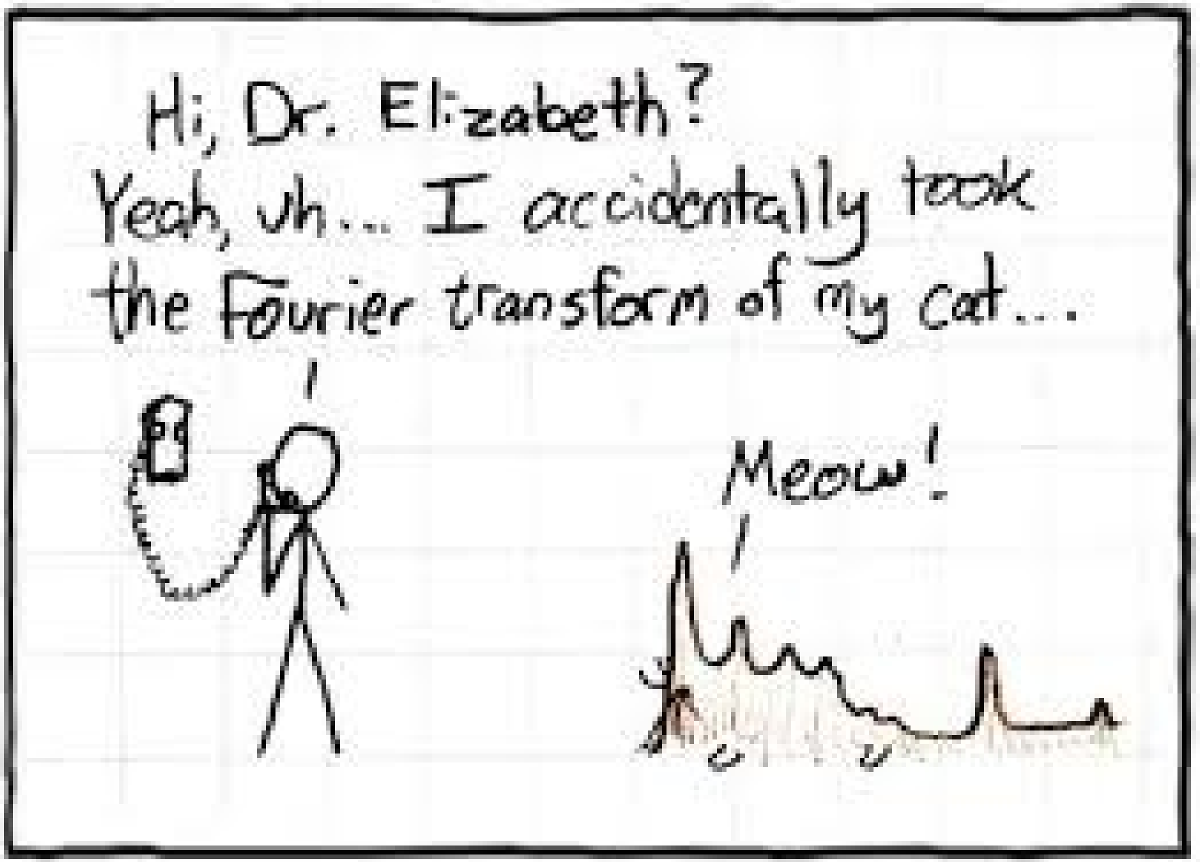
Joe Callingham (ASTRON)

Kenyan Radio Astronomy School,
Nairobi, Kenya
30th of May 2018

Thanks to Jack Radcliffe, Anna Scaife, and David Wilner

DARA
DEVELOPMENT IN AFRICA WITH RADIO ASTRONOMY

UNIVERSITY OF LEEDS

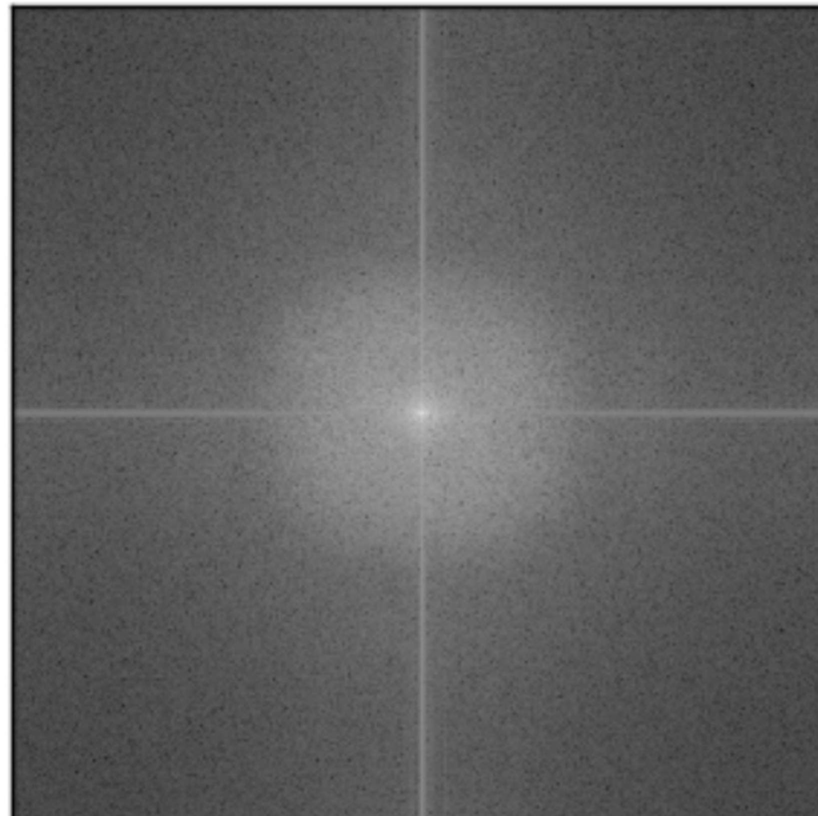

Hi, Dr. Elizabeth?
Yeah, uh... I accidentally took the Fourier transform of my cat...
Meow!

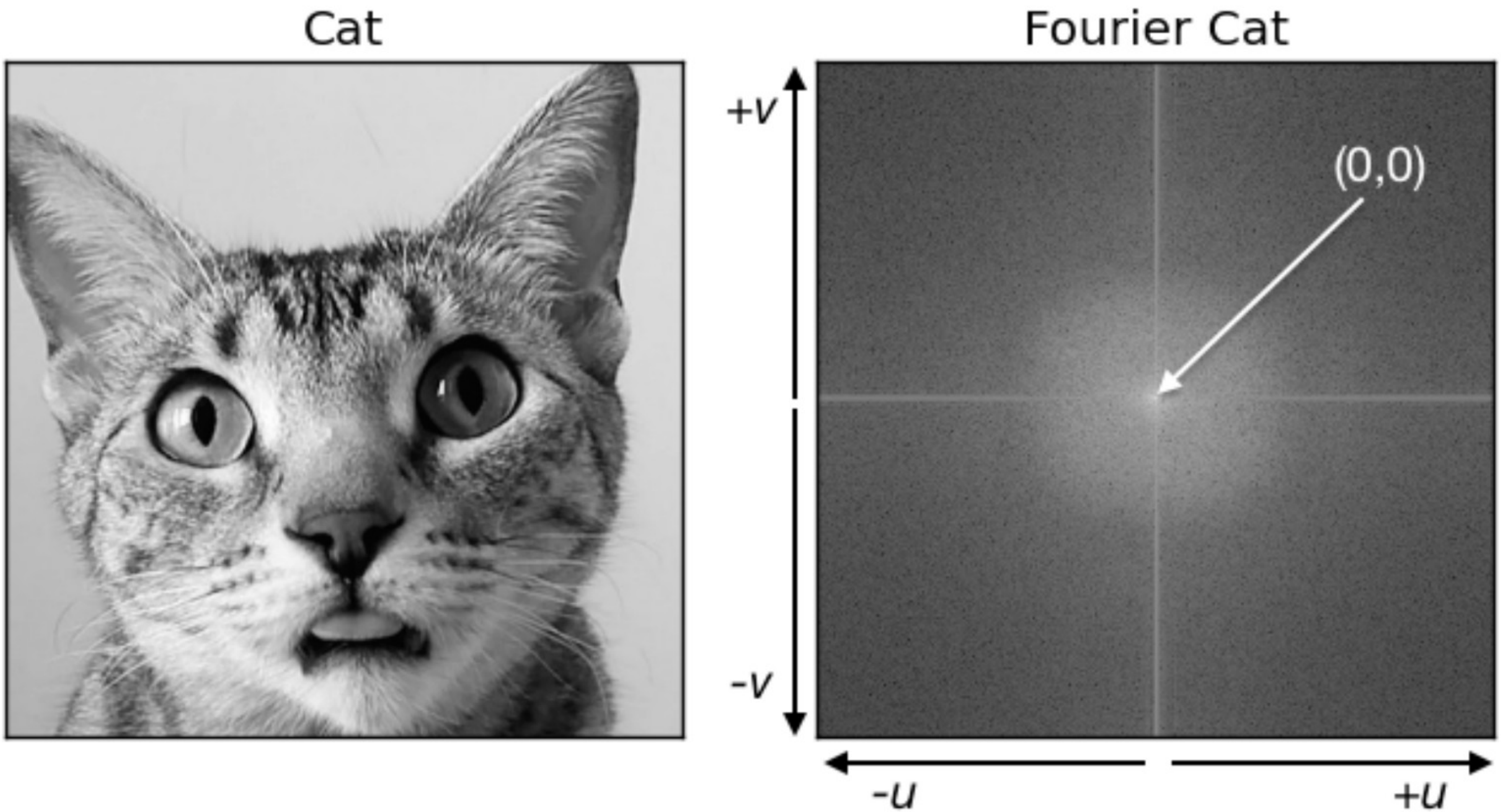
ASTRON

Disclaimer: This is not my cat.

Cat

Fourier Cat

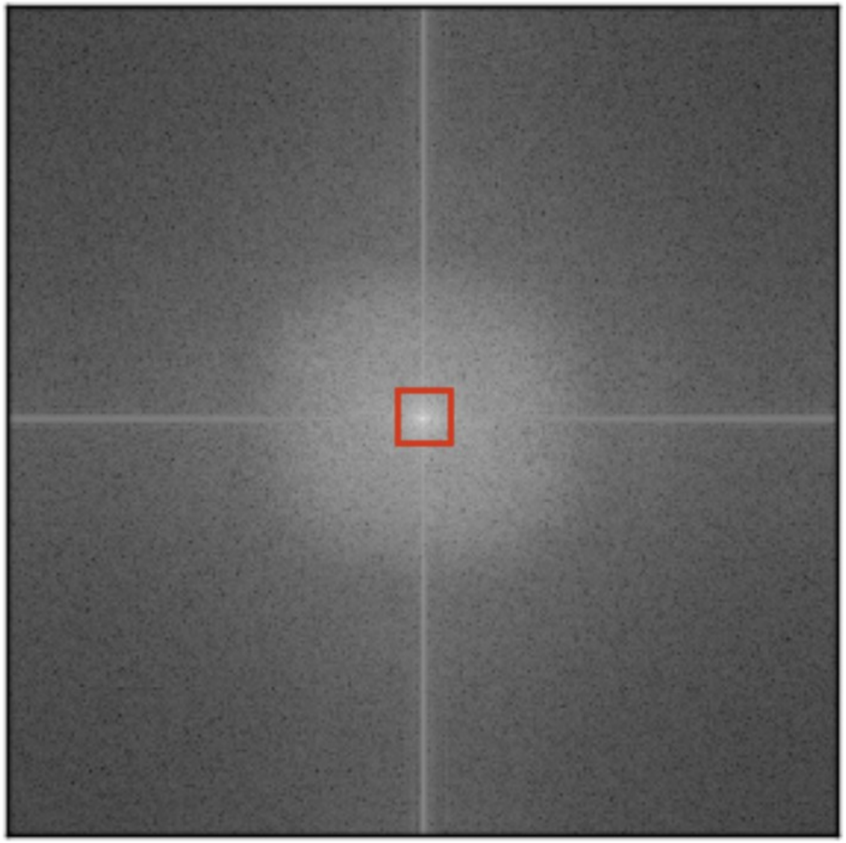




Cat
Fourier Cat
+v
-v
(0,0)
-u
+u

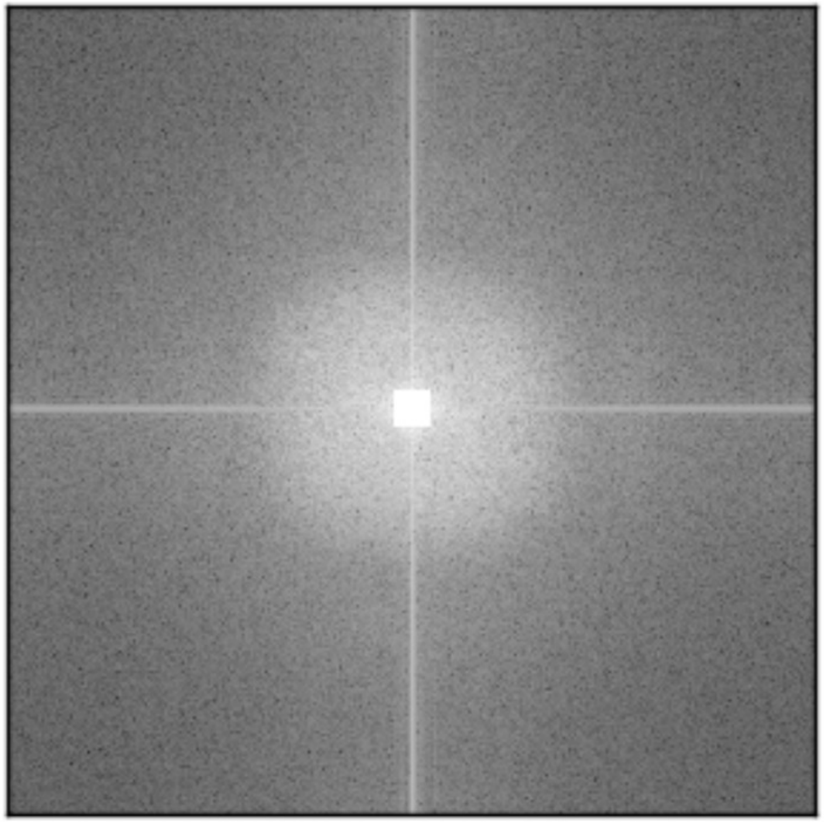
Cat

Fourier Cat

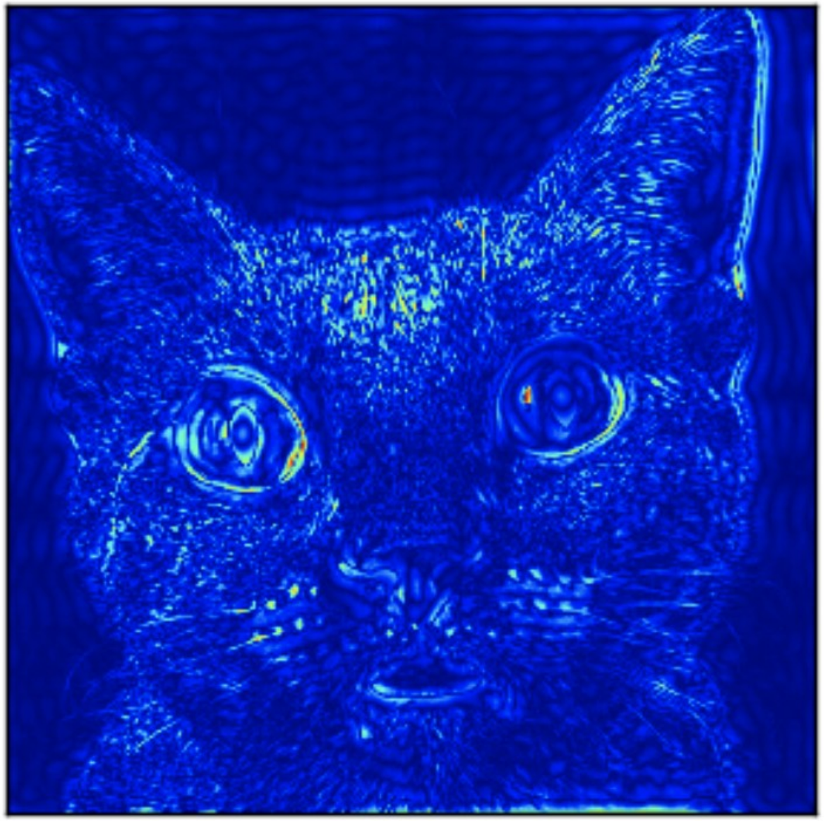


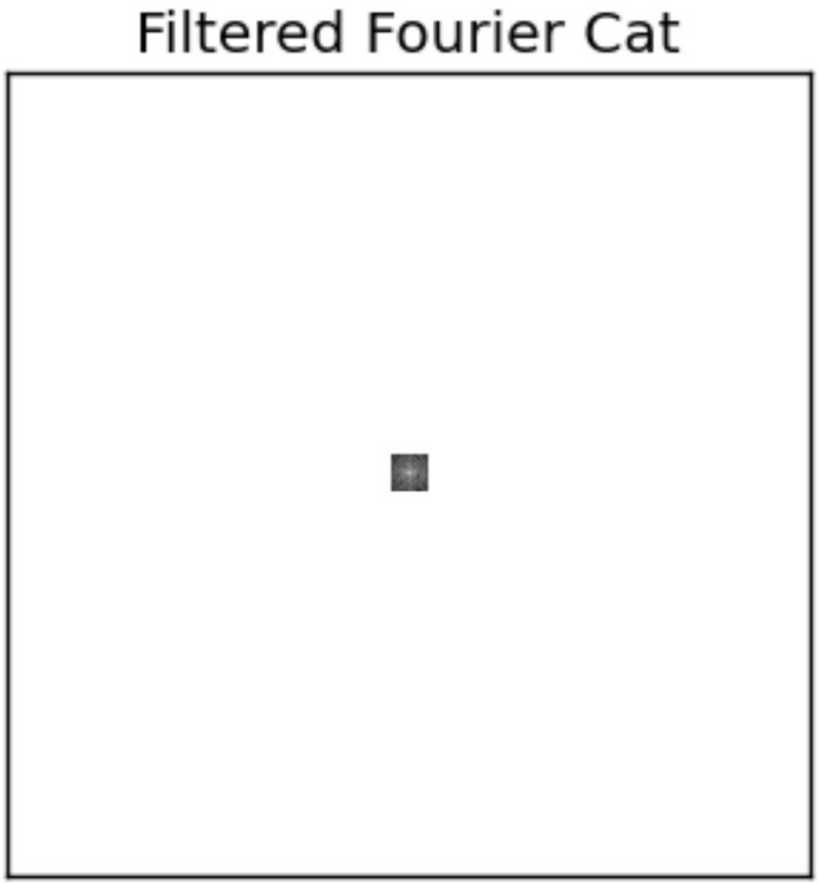
ASTRON

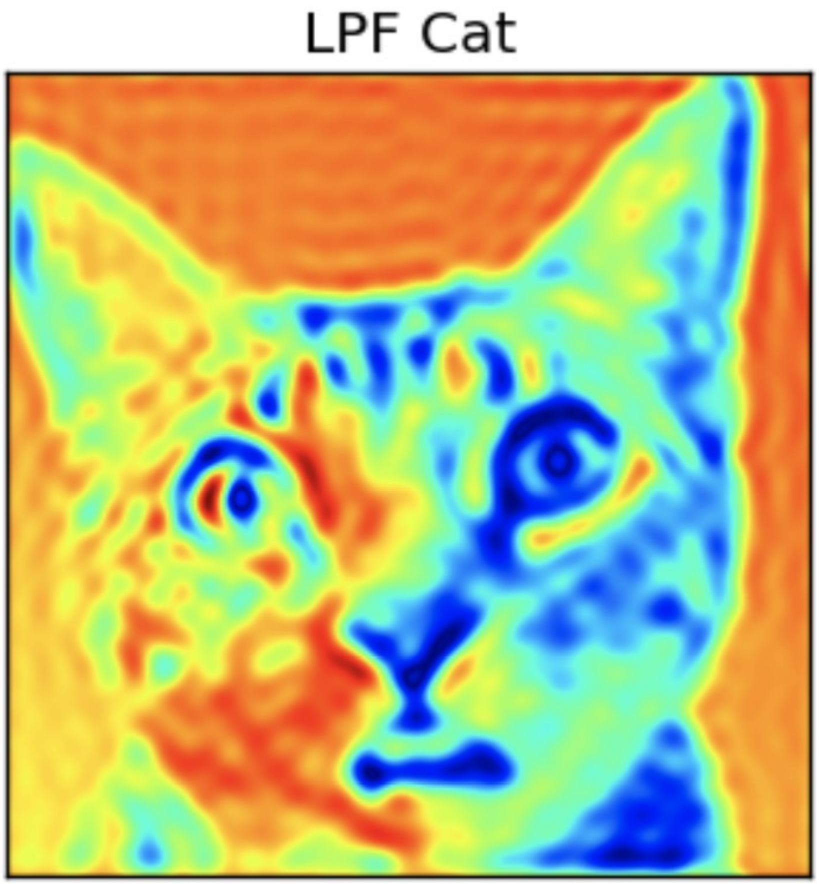
Filtered Fourier Cat



HPF Cat




Filtered Fourier Cat

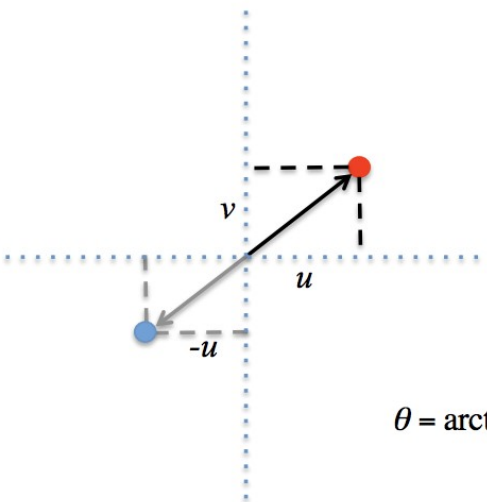

LPF Cat

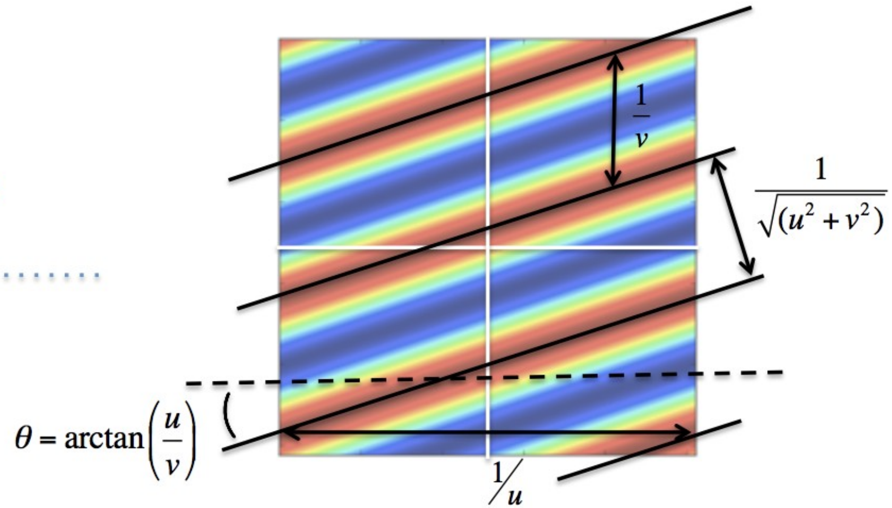
Small Fourier Frequencies = Large Scale Image Structure
Large Fourier Frequencies = Small Scale Image Structure

Big is Small & Small is Big

Visibilities

FOURIER COMPONENTS



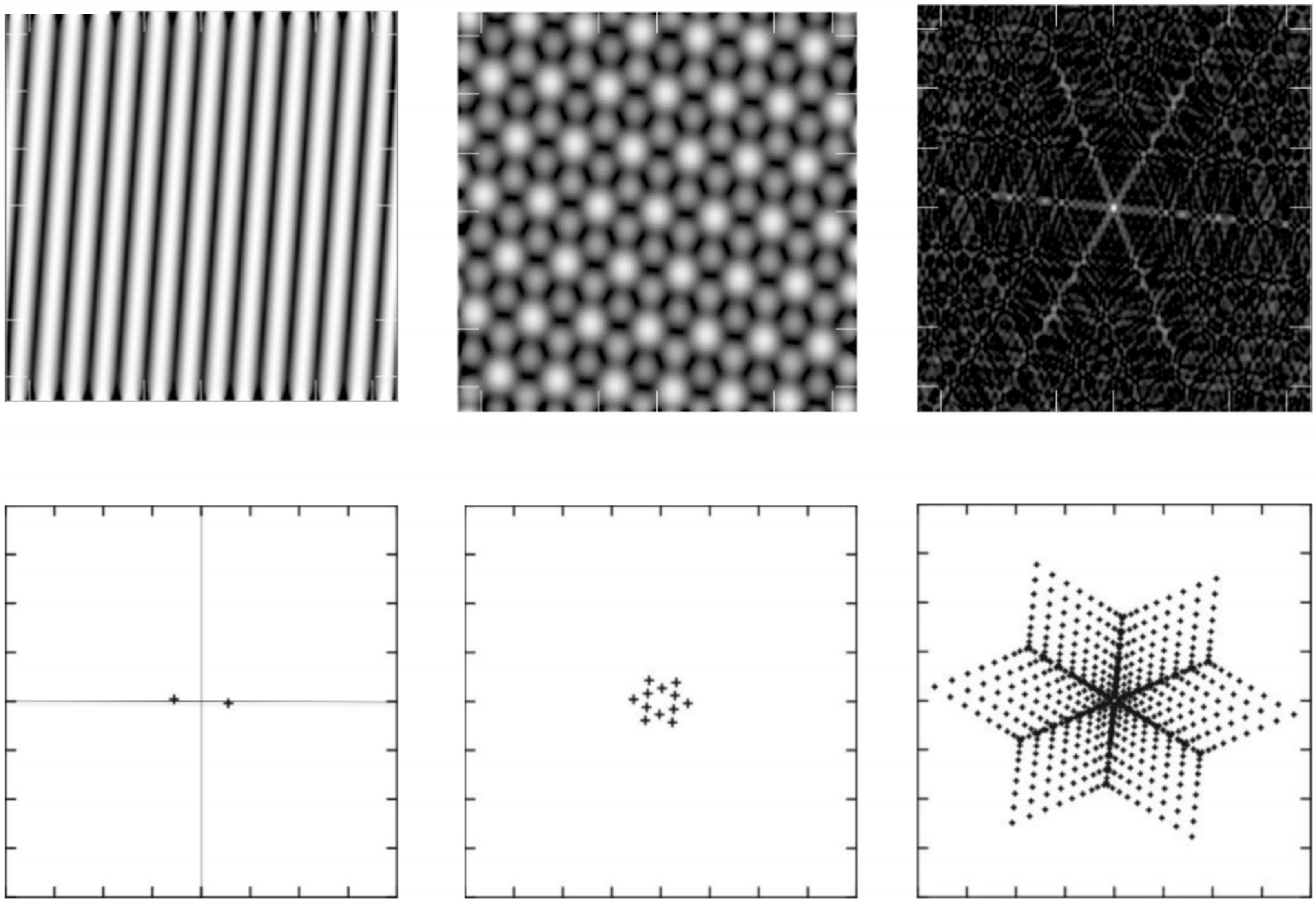


Writing the equation in this way allows us to visualise how our image is composed.

$$I_{meas}(l,m) = \frac{1}{M}\sum_{i=1}^{M} A(u_i,v_i)\cos[2\pi(u_i l + v_i m) + \phi_i]$$

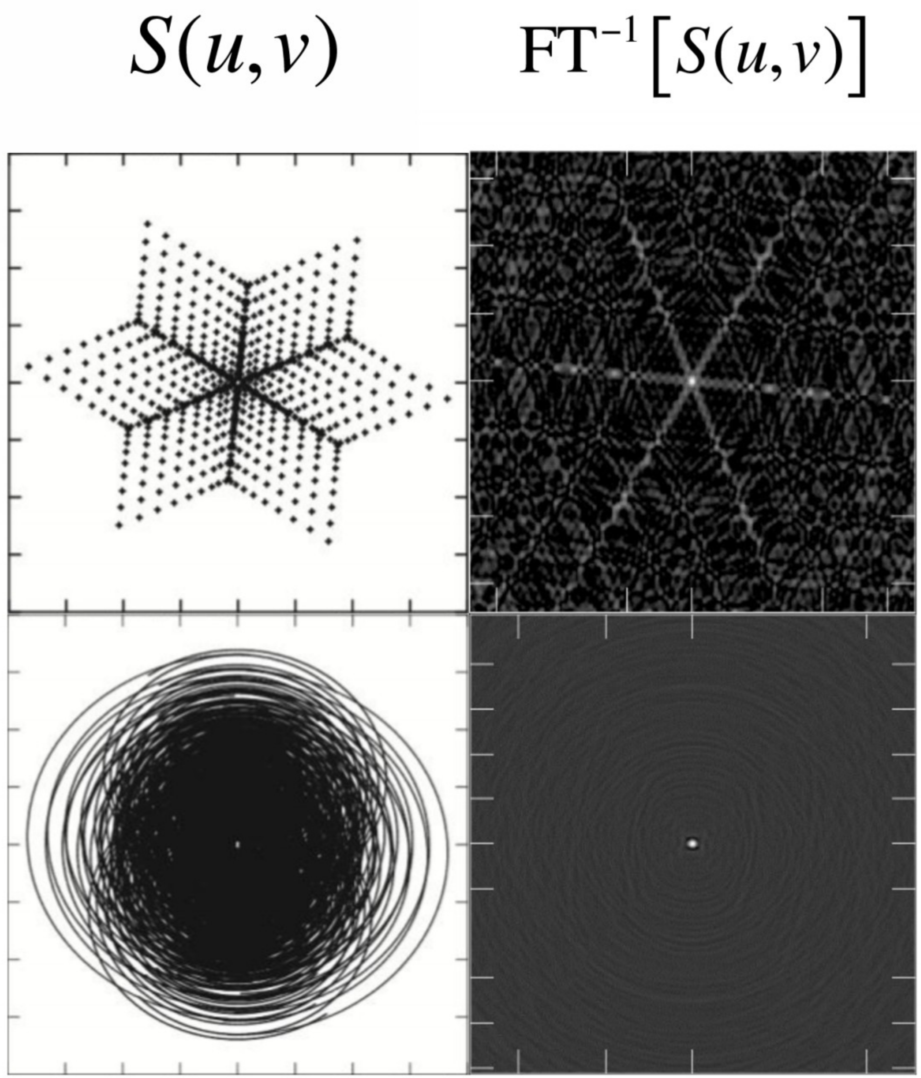
SYNTHESISED BEAM

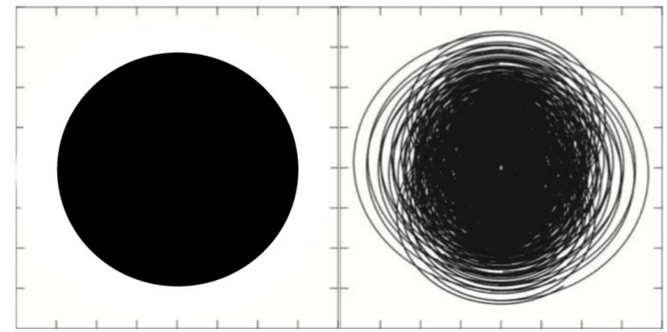
$\mathrm{FT}^{-1}\left[S(u,v)\right]$



$S(u,v)$

SYNTHESISED BEAM





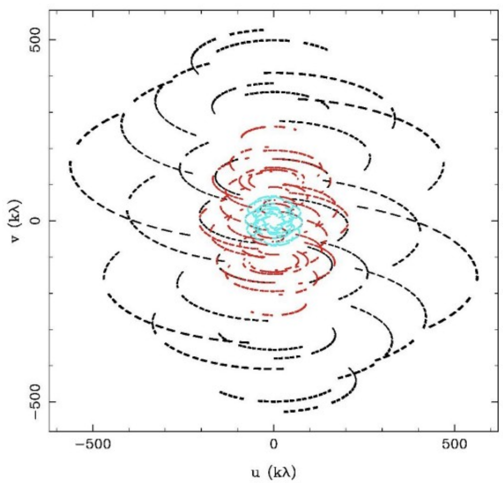
The baseline (uv) sampling defines the measured angular scales and sets the resolution.

<u>Disambiguation:</u>

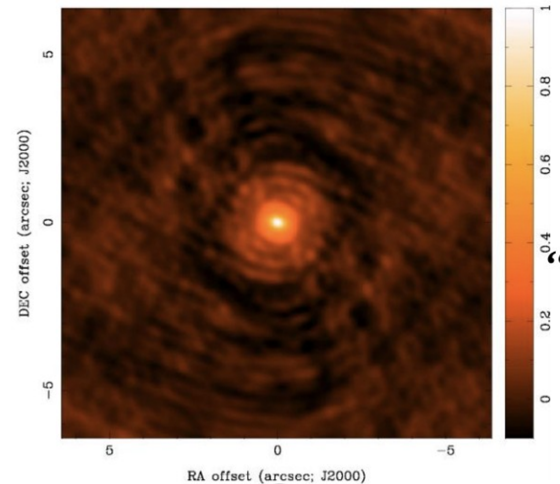
Synthesized beam
= point spread function
= dirty beam

Dirty Beam and Dirty Image

$S(u,v)$



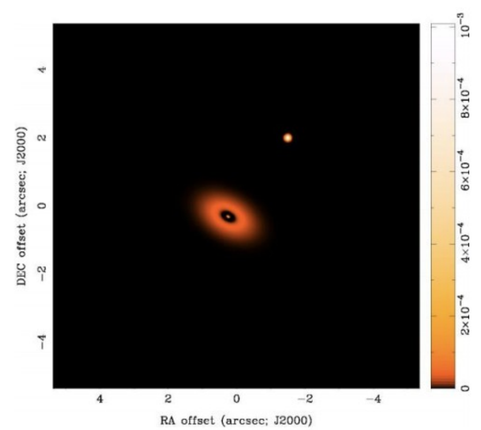
$\xrightarrow{\mathcal{F}}$



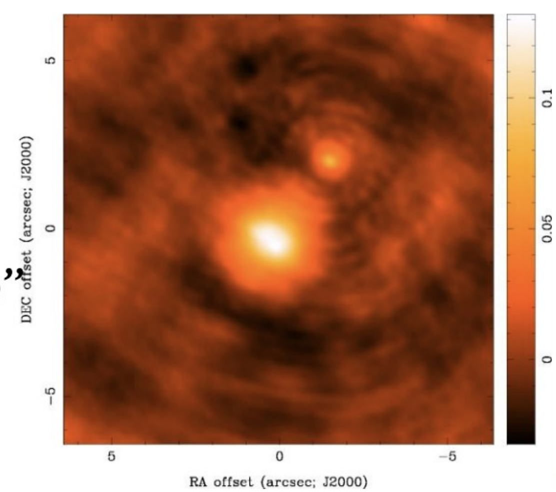
$s(l,m)$
"dirty beam"

$*$

$T(l,m)$



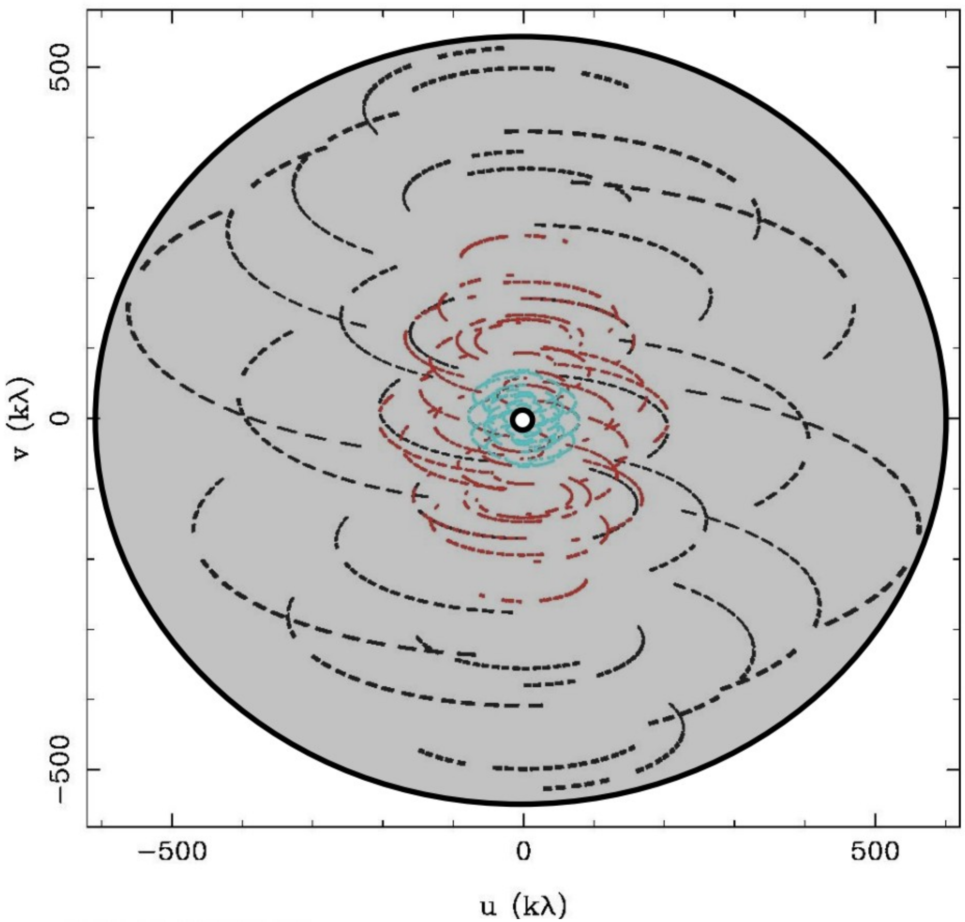
$\longleftarrow$



$T^D(l,m)$
"dirty image"

IMPLICATIONS OF UV SAMPLING

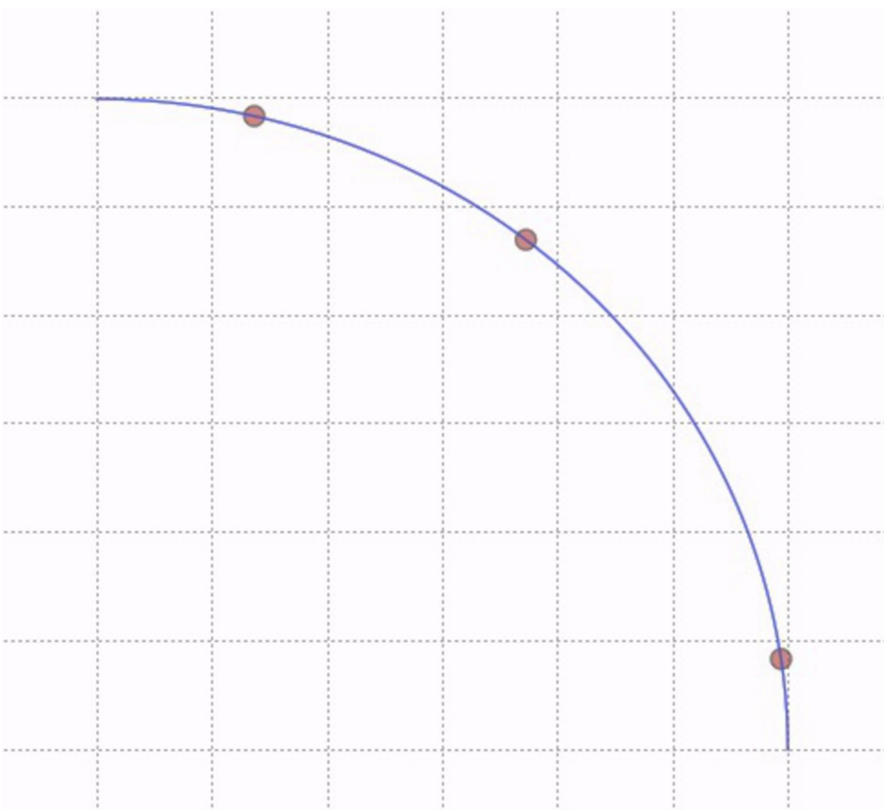
samples of *V(u,v)* are limited by number of antennas and by Earth-sky geometry



- *outer boundary*
  - no information on smaller scales
  - resolution limit
- *inner hole*
  - no information on larger scales
  - extended structures invisible
- *irregular coverage between boundaries*
  - sampling theorem violated
  - information missing

GRIDDING

FFTs are faster but they also introduce complications.



FFTs require regularly spaced (u,v) data.

Interferometer data can be regularly spaced in time and frequency, but are not regularly spaced in u and v.

In order to use an FFT we need to GRID our data. This causes its own issues…

DECONVOLUTION

The basic operation of an (ideal) interferometer baseline measures (small sky approximation, w→0):

$$V(u,v) \approx \iint I(l,m) e^{-2\pi i(ul+vm)} dl dm$$

We can, in principle, measure I(l,m) for all u,v. We can then use a Fourier transform to recover the sky brightness distribution:

$$I(l,m) \approx \iint V(u,v) e^{2\pi i(ul+vm)} du dv$$

However V(u,v) is not known everywhere but is sampled at particular places on the u-v plane

Nb: (l,m,n) notation is essentially the same as (x,y,z) coordinates used in the prev. talks

Nb: notation is essentially the same as coordinates used in the prev. talks

DECONVOLUTION

ASTRON

This sampling function can be described by S(u,v) and is equal to 1 when the uv plane is sampled and zero otherwise:

$$I^D(l,m) = \iint V(u,v)S(u,v)e^{2\pi i(ul+vm)}dudv$$

the Fourier transform of the sampled visibilities yields the true sky brightness convolved with the point spread function

$$I^D(l,m) = I(l,m) * B$$

Where *B* is known as the 'dirty beam' or the 'point spread function' and is the FT of the sampling function.

$$B(l,m) = \iint S(u,v)e^{2\pi i(ul+vm)}dudv$$

CASA IMAGE CONSTRUCTION

ASTRON

```
#  clean :: Invert and deconvolve images with selected algorithm
vis                 =         ''        #  Name of input visibility file
imagename           =         ''        #  Pre-name of output images
outlierfile         =         ''        #  Text file with image names, sizes,
                                        #   centers for outliers
field               =         ''        #  Field Name or id
spw                 =         ''        #  Spectral windows e.g. '0~3', '' is
                                        #   all
selectdata          =       True        #  Other data selection parameters
     timerange      =         ''        #  Range of time to select from data
     uvrange        =         ''        #  Select data within uvrange
     antenna        =         ''        #  Select data based on antenna/baseline
     scan           =         ''        #  Scan number range
     observation    =         ''        #  Observation ID range
     intent         =         ''        #  Scan Intent(s)

mode                =      'mfs'        #  Spectral gridding type (mfs, channel,
                                        #   velocity, frequency)
     nterms         =          1        #  Number of Taylor coefficients to
                                        #   model the sky frequency dependence
     reffreq        =         ''        #  Reference frequency (nterms > 1),''
                                        #   uses central data-frequency
gridmode            =         ''        #  Gridding kernel for FFT-based
                                        #   transforms, default='' None
niter               =        500        #  Maximum number of iterations
gain                =        0.1        #  Loop gain for cleaning
threshold           =   '0.0mJy'        #  Flux level to stop cleaning, must
                                        #   include units: '1.0mJy'
psfmode             =    'clark'        #  Method of PSF calculation to use
                                        #   during minor cycles
imagermode          =  'csclean'        #  Options: 'csclean' or 'mosaic', '',
                                        #   uses psfmode
     cyclefactor    =        1.5        #  Controls how often major cycles are
                                        #   done. (e.g. 5 for frequently)
     cyclespeedup   =         -1        #  Cycle threshold doubles in this
                                        #   number of iterations

multiscale          =         []        #  Deconvolution scales (pixels); [] =
                                        #   standard clean
interactive         =      False        #  Use interactive clean (with GUI
                                        #   viewer)
mask                =         []        #  Cleanbox(es), mask image(s),
                                        #   region(s), or a level
imsize              = [256, 256]        #  x and y image size in pixels.  Single
                                        #   value: same for both
cell                = ['1.0arcsec']     #  x and y cell size(s). Default unit
                                        #   arcsec.
phasecenter         =         ''        #  Image center: direction or field
                                        #   index
restfreq            =         ''        #  Rest frequency to assign to image
                                        #   (see help)
stokes              =        'I'        #  Stokes params to image (eg
                                        #   I,IV,IQ,IQUV)
weighting           =  'natural'        #  Weighting of uv (natural, uniform,
                                        #   briggs, ...)
uvtaper             =      False        #  Apply additional uv tapering of
                                        #   visibilities
modelimage          =         ''        #  Name of model image(s) to initialize
                                        #   cleaning
restoringbeam       =       ['']        #  Output Gaussian restoring beam for
                                        #   CLEAN image
pbcor               =      False        #  Output primary beam-corrected image
minpb               =        0.2        #  Minimum PB level to use
usescratch          =      False        #  True if to save model visibilities in
                                        #   MODEL_DATA column
allowchunk          =      False        #  Divide large image cubes into channel
                                        #   chunks for deconvolution
```

- clean is the CASA imaging routine

- To achieve a basic image, need to set:
  - vis - your data
  - imagename
  - niter - no. of CLEAN iterations (next slide)
  - imsize - size of the image in pixels (needs to be as small as possible to decrease computation time)
  - cell - angular extent of each pixel (need to adequately sample the psf)

  Rule of thumb:

$$\mathrm{cell} \sim \lambda_\mathrm{f}/3B$$

λ_f - wavelength of highest frequency channel

B - longest baseline length

DECONVOLUTION

To recover the real brightness distribution we just need to deconvolve… easier said than done:

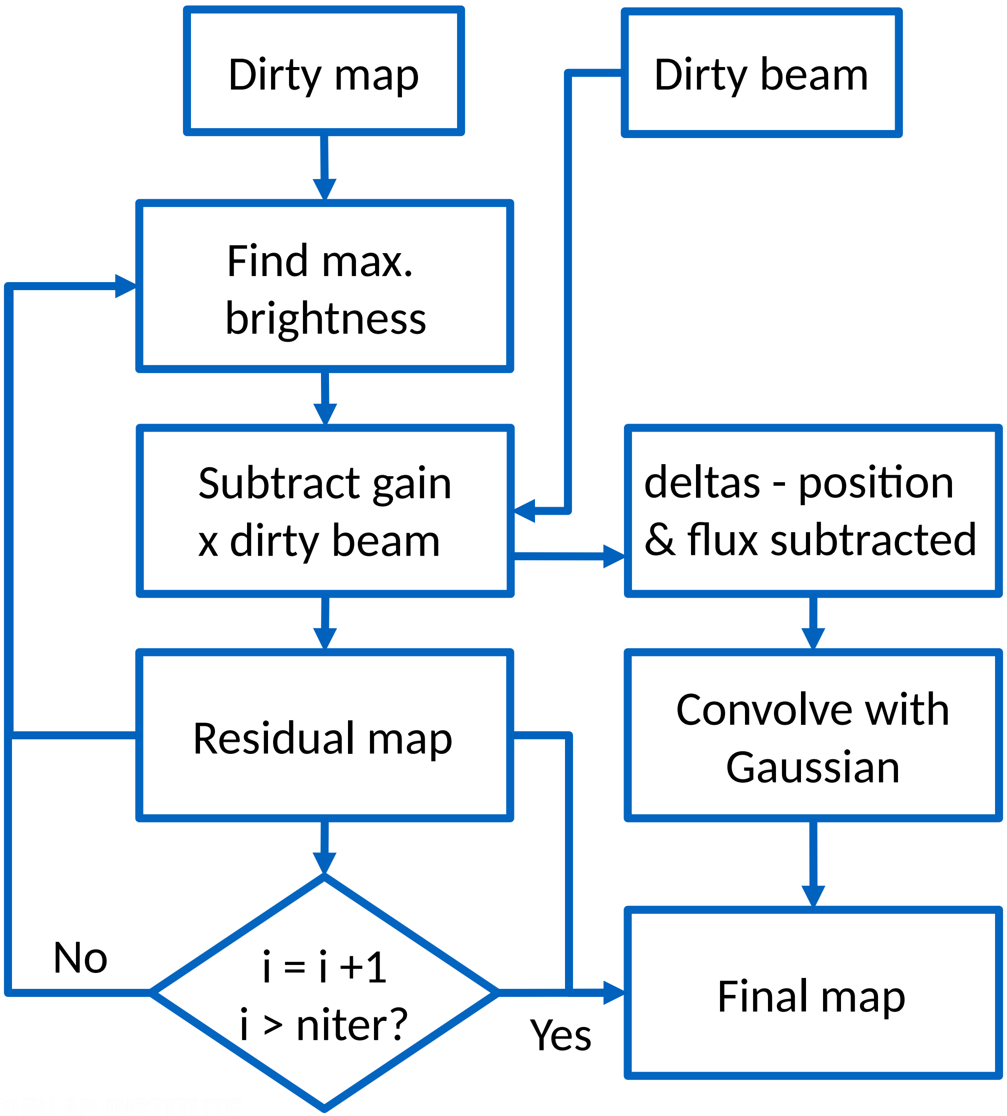
- A vast number of images are consistent with the data inc. the dirty beam.

- We need to take a Bayesian approach - supply priors (i.e. extra information/ assumptions) so we can find the most probable brightness distribution.

- Simplest scheme (but not only): Sky is mostly empty and consists of a finite number of unresolved point sources.

→ The basis of the Hogbom CLEAN algorithm (1974)

HOGBOM CLEAN & VARIANTS



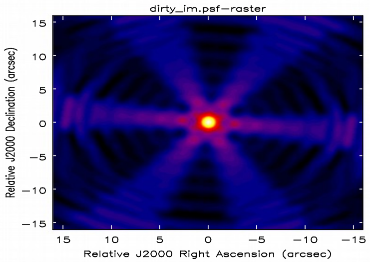
- initialize
  - a *residual map* to the dirty map
  - a *Clean Component* list

1. identify the highest peak in the residual map as a point source
2. subtract a fraction of this peak from the *residual map* using a scaled dirty beam, *s(l,m)* x gain
3. add this point source location and amplitude to the *Clean Component* list
4. goto step 1 (an iteration) unless stopping criterion reached

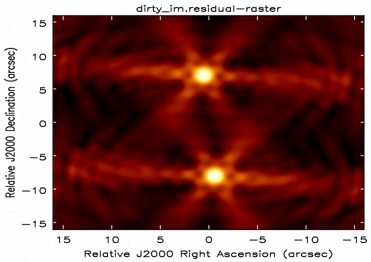
CLEAN DECONVOLUTION

JVLA simulation, 2hr observation targeting two 0.1 Jy point sources + some phase corruption included

Dirty beam



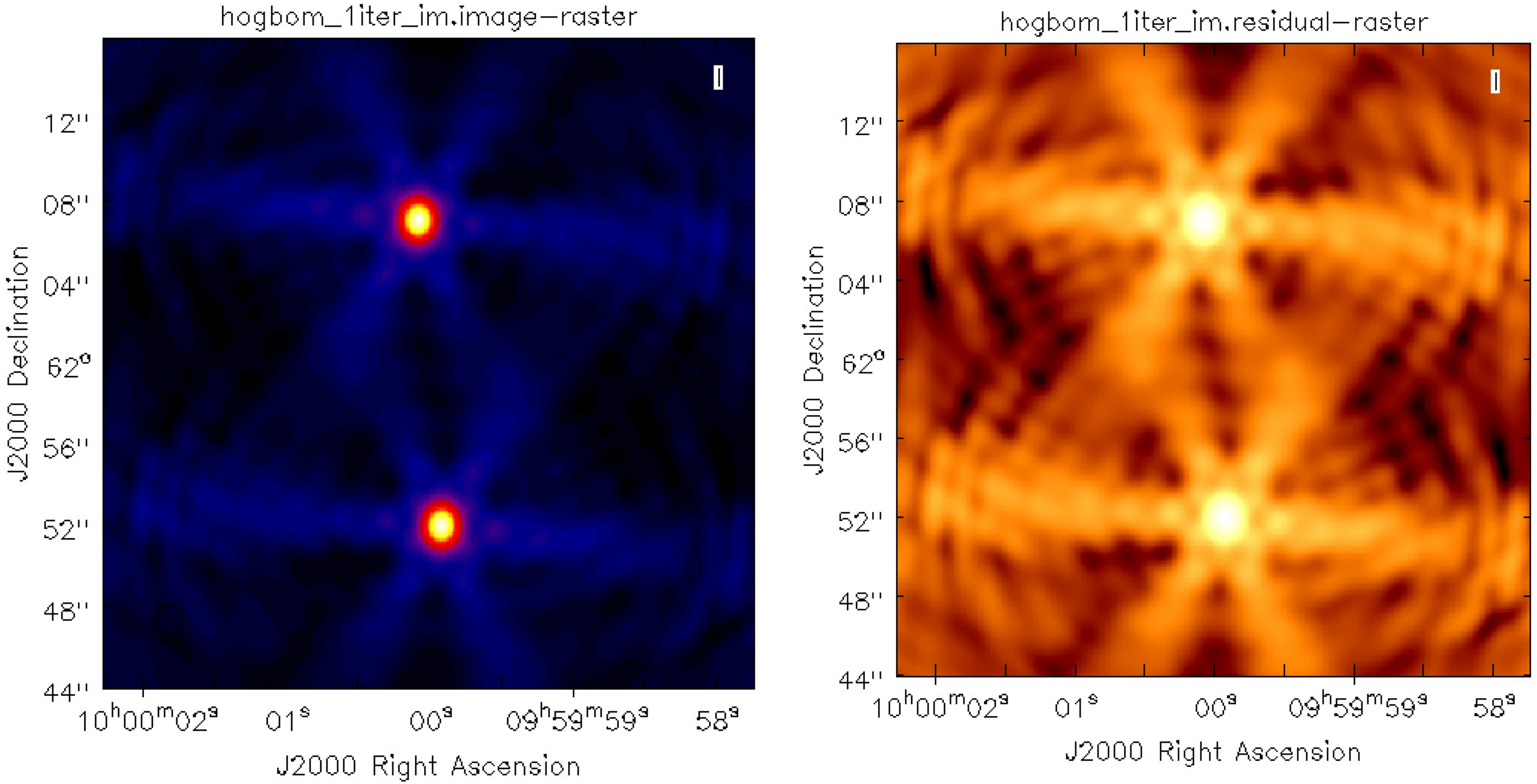
Dirty image



CLEAN DECONVOLUTION

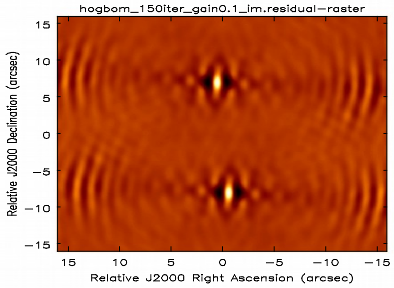
Hogbom CLEAN

Image & residual after 1 iteration with 0.5 gain

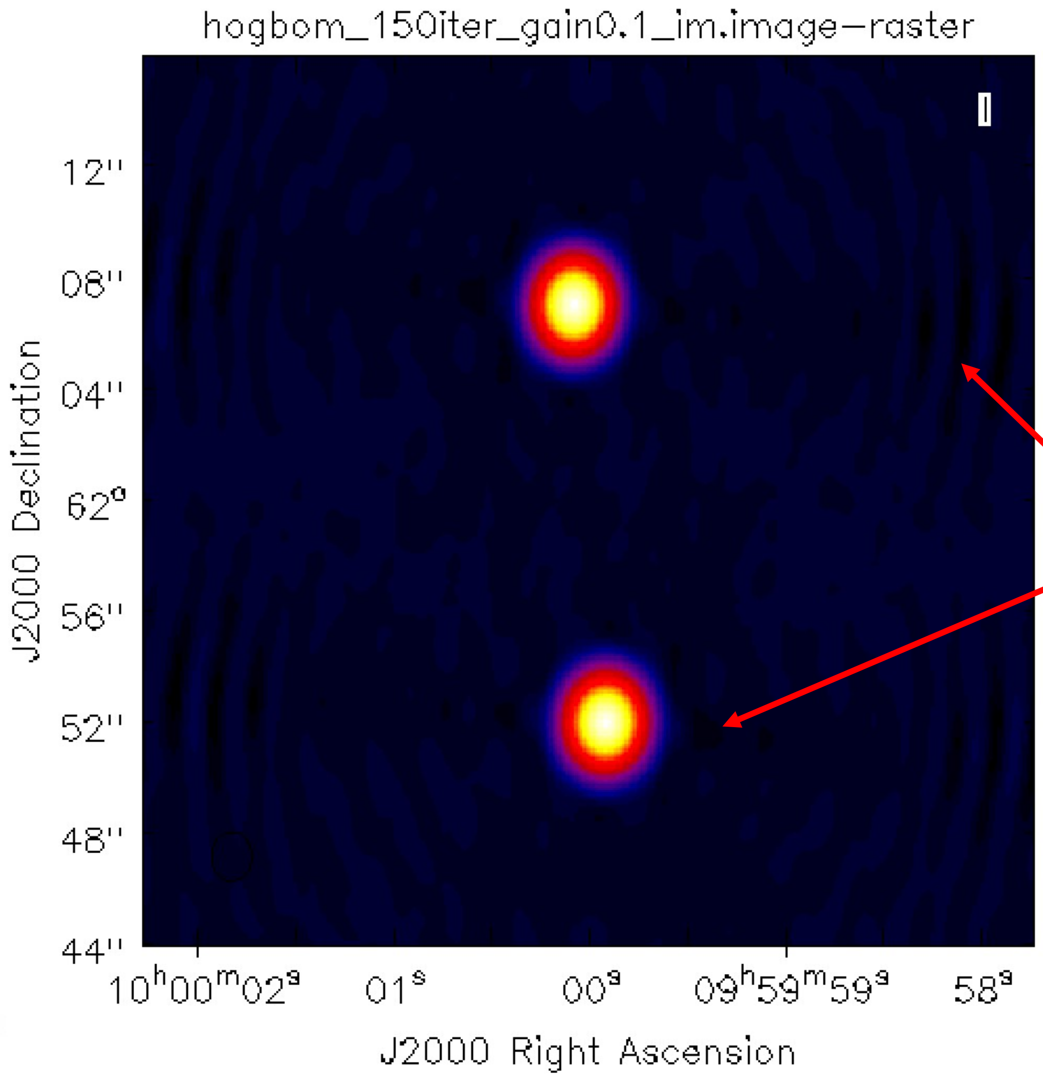


Hogbom CLEAN

Residual after 150 iterations with 0.1 gain



CLEAN map (residual+CLEAN components) after 150 iterations



Some artefacts left from deconvolution

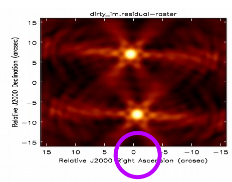
CLEAN DECONVOLUTION

CLEAN is far from perfect, but we can lend it a hand:

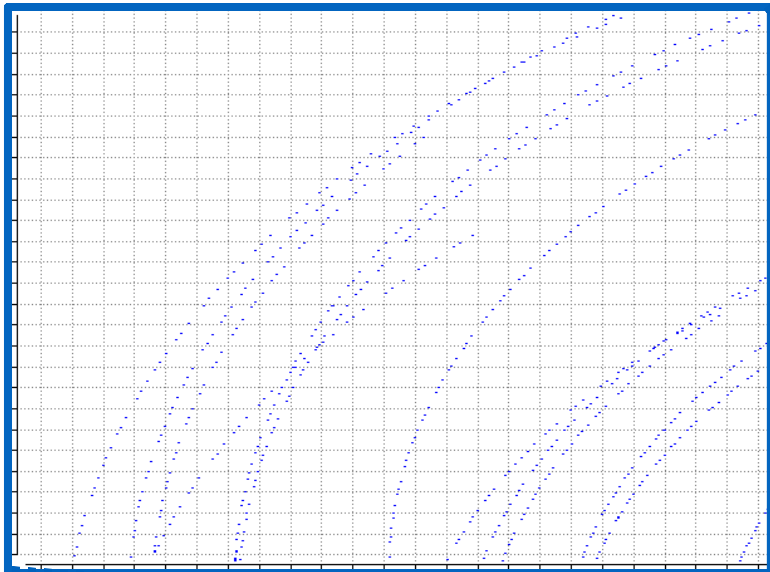
CLEAN consists of two 'cycles':

I. Minor cycles - subtract subimages of the dirty beam
II. Major cycles - Fourier Transform residual map and subtract

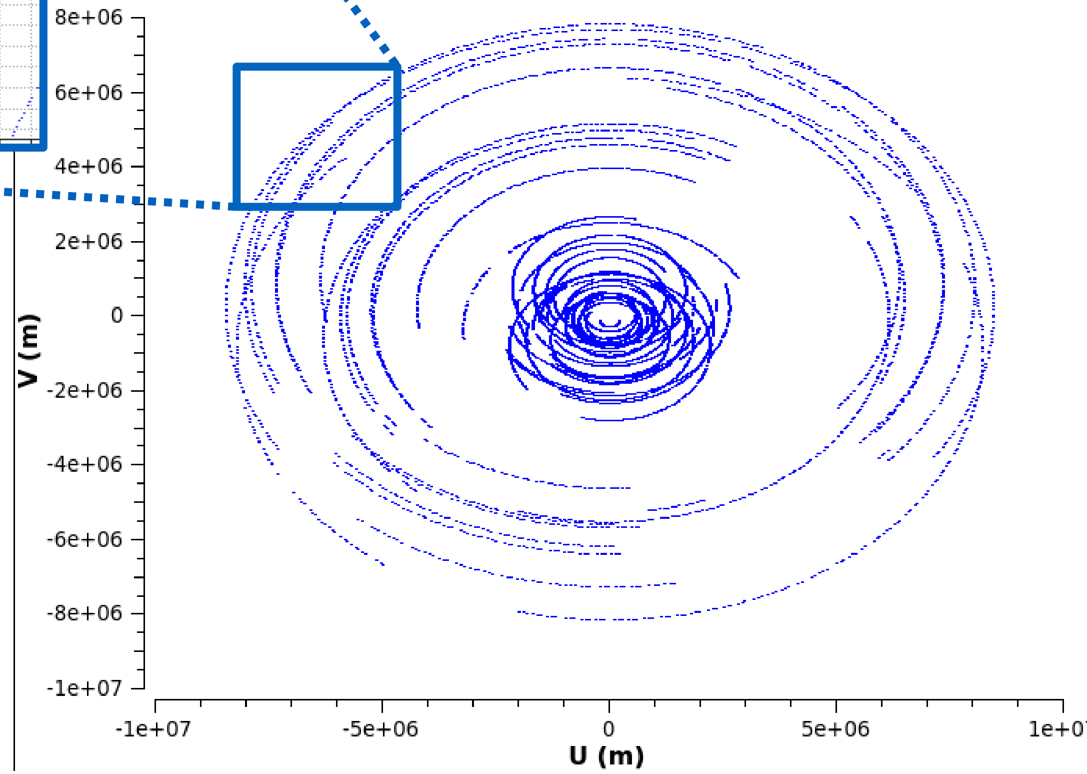
We can use windowing to tell the algorithm where the flux lies.
This should be used when you **know** the flux you see is real!



2. WEIGHTING

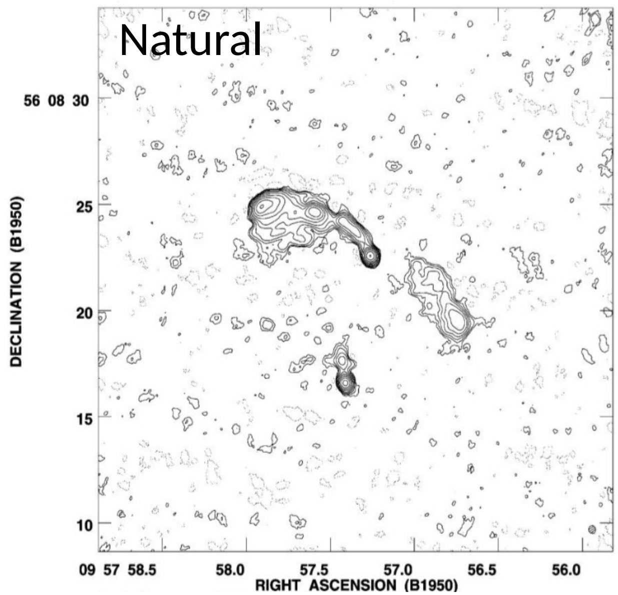


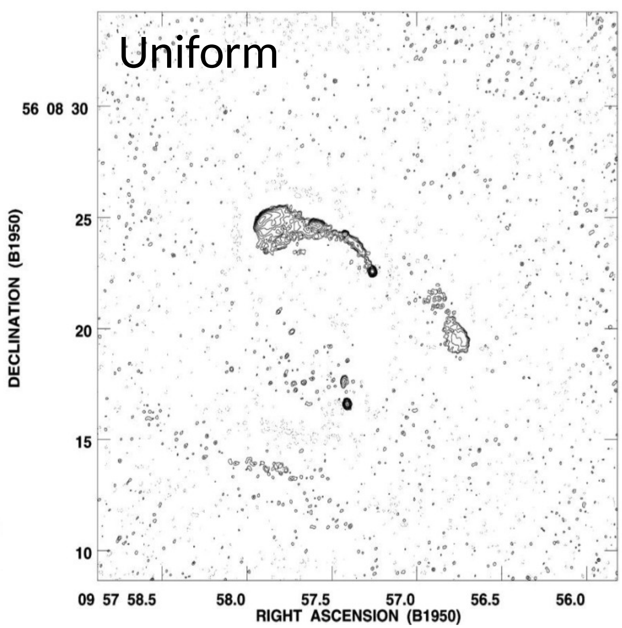
Integrations are distributed over a greater number of sampled grid points in the outer uv plane than the inner regions



- Data interpolated on 2^n grid
- Weights unmodified by local density - ‘Natural’
- Weights divided by local density of points - ‘Uniform’

UV WEIGHTING





Natural weighted images have low spatial frequencies are weighted up (due to gridding) and gives:

- Best S/N
- Worse resolution

Uniform weighted images low have spatial frequencies weighted down and the data are not utilised optimally (may be subject to a deconvolution striping instability) resulting in:

- Worse S/N
- Best resolution

Compromises exist:

- Briggs (robust) weighting parameter -5 to +5. (next slide)

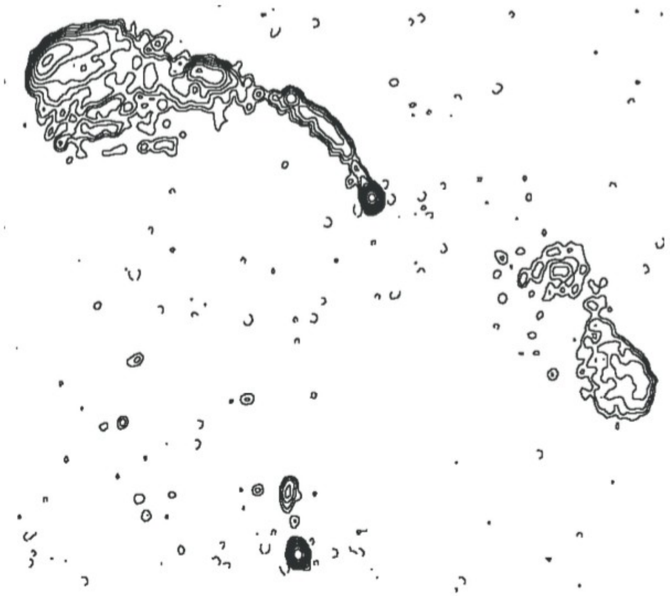
Implementation in CASA clean

```
weighting          =  'natural'        #  Weighting of uv (natural, uniform,
                                       #   briggs, ...)
```

UV WEIGHTING: 'BRIGGS WEIGHTING'

- Originally derived as a cure for striping – Natural weighting is immune and therefore most 'robust'

Robust 0 image



- Varies effective weighting as a function of local u-v weight density
  - Where weight density is low – effective weighting is natural
  - Where weight density is high – effective weighting is uniform

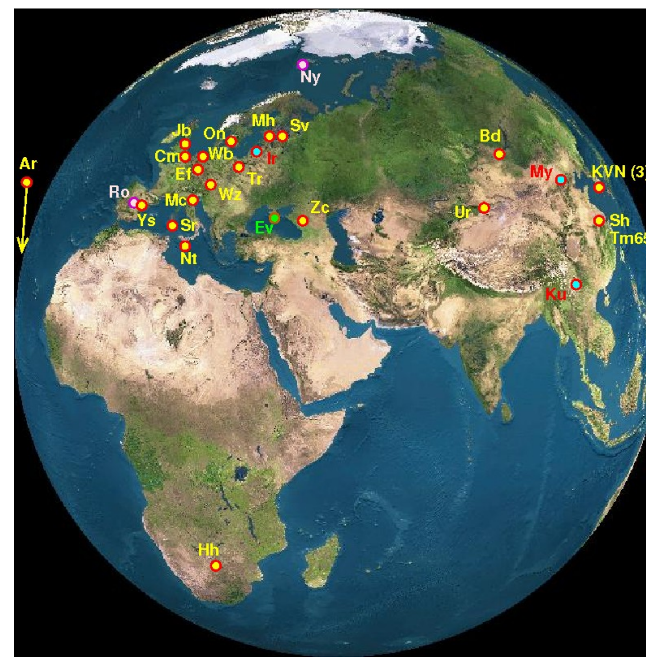
- Modifies the variations in effective weight found in uniform weighting → more efficient use of data & lower thermal noise

- ROBUST = – 5 is nearly pure uniform ROBUST = + 5 is nearly pure natural
  ROBUST = 0 is a good compromise (Contoured)

- Can produce images close to uniform weighting resolution with noise levels close to natural weighting. See CASA webpage for other weighting schemes!

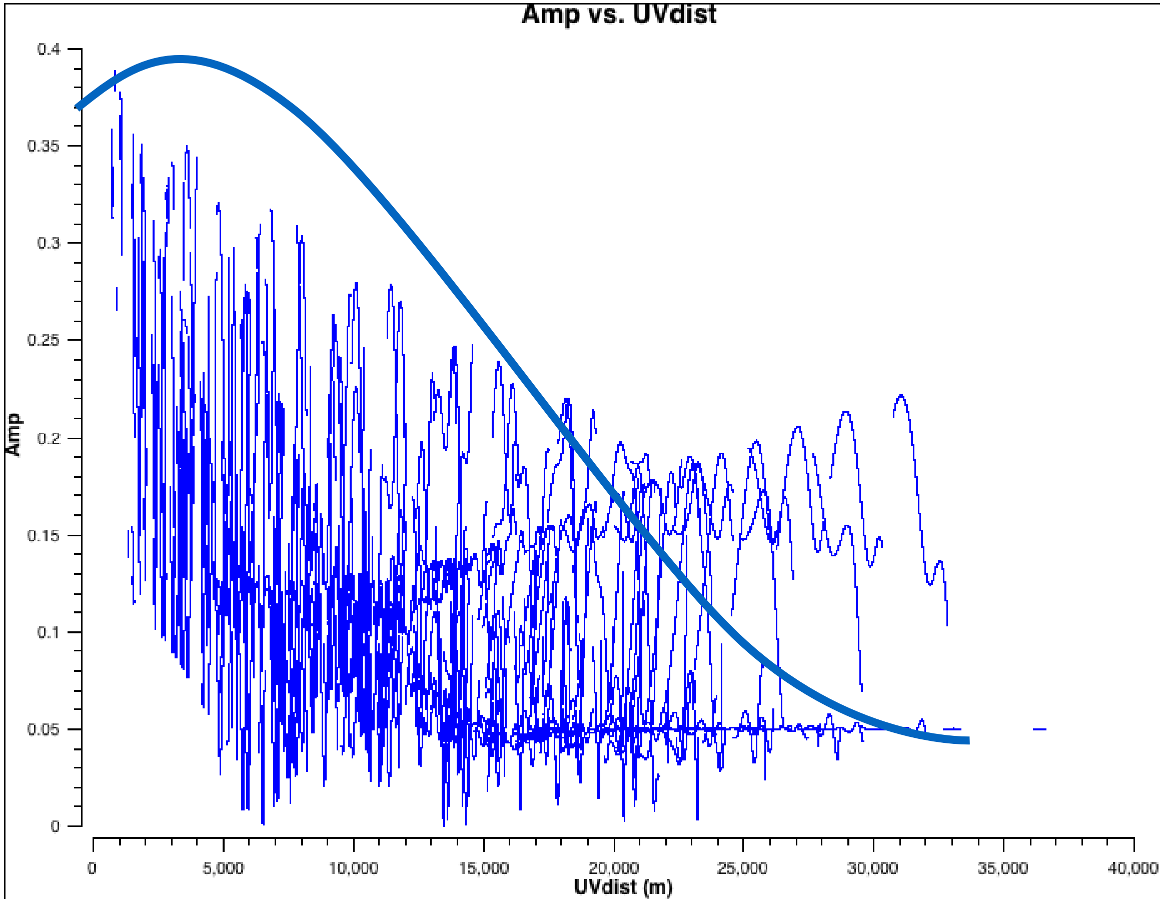
WEIGHTING BY TELESCOPE

- Many arrays are heterogeneous e.g. e-MERLIN, EVN & AVN (when built)
- To get the best S/N need to increase weighting on larger telescopes so they contribute more.
- Nb. this can change the resolution depending on the baseline distribution.



UV TAPERING

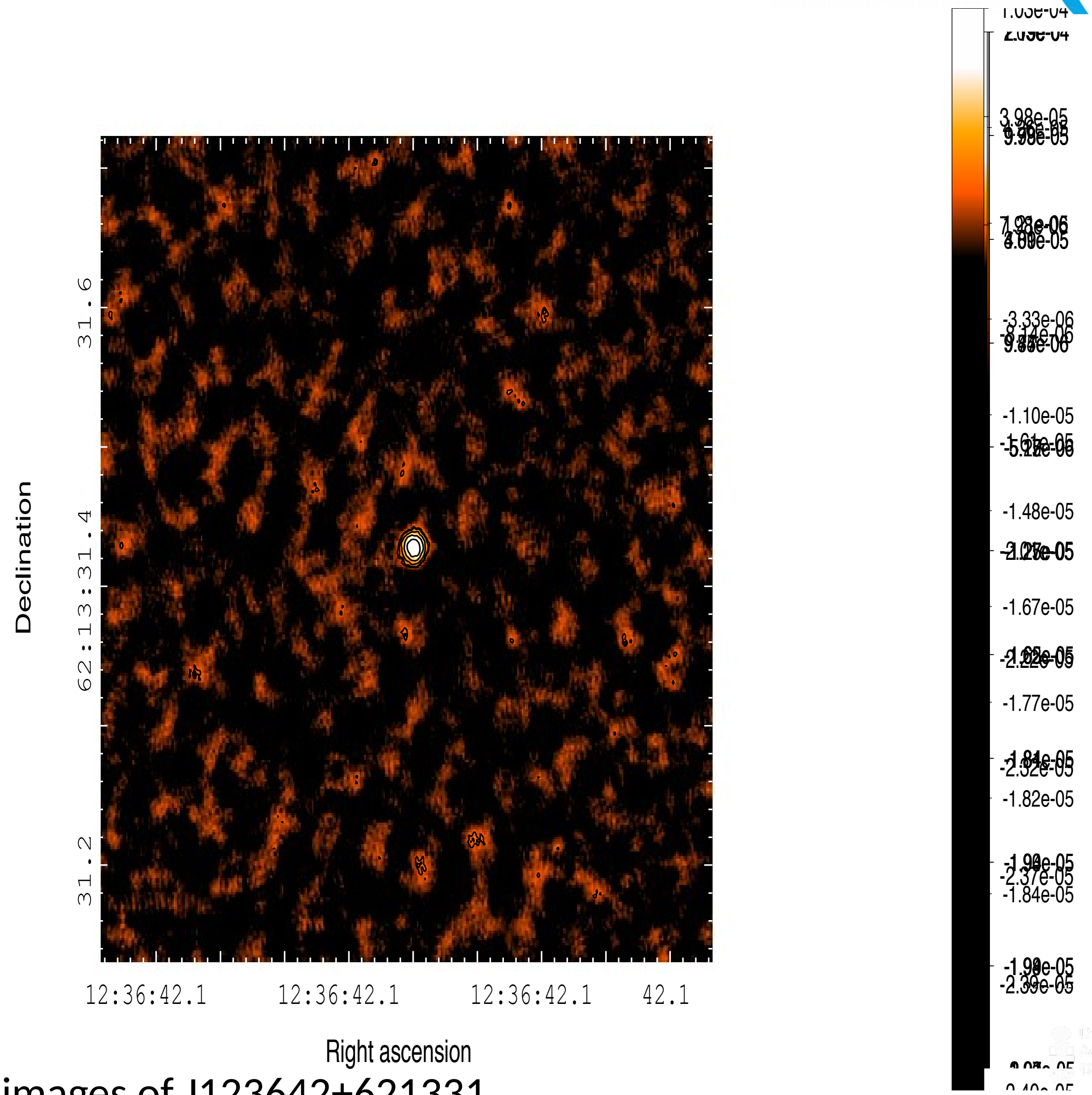
Gaussian u-v taper or u-v range can smooth the image but at the expense of sensitivity since data are excluded or data usage is non-optimal



Can compromise image quality in VLBI arrays by severely restricting the *u-v* coverage

Controlled by the uvtaper parameter in CASA task tclean/clean

UV TAPERING



eMERLIN + EVN images of J123642+621331

3. WIDE-FIELD IMAGING

A 'wide-field' image is defined as:

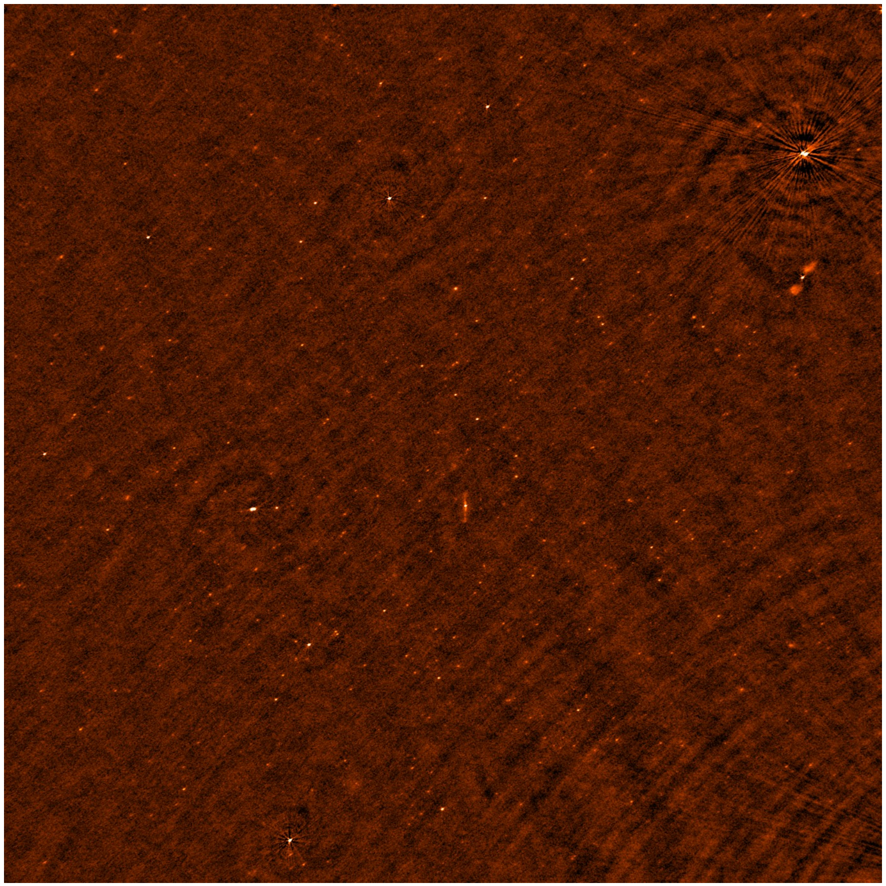
- An image with large numbers of resolution elements across them
- Or multiple images distributed across the interferometer primary beam

In order to image the entire primary beam you have to consider the following distorting effects:

1. Bandwidth smearing
2. Time smearing
3. Non-coplanar baselines (or the 'w' term) - Covered in advanced imaging
4. Primary beam response

CONFUSION

ASTRON



JVLA image of GOODS-N showing confusion from a 0.25Jy source to the SE

- Bright radio sources on the edge of the primary beam give rise to ripples in the centre of the field of view

- The primary beam is spectrally dependent, so image subtraction should include such corrections and be performed in full spectral-line mode

- Pointing errors introduce gain and phase changes on the edge of the primary beam. If severe, the apparent source structure may change – attempt multiple snapshot subtraction on timescales comparable with pointing error change

CONFUSION

So how do we deal with these sources?

1. Outlier fields (the CASA default option) - deconvolve the confusing source while imaging the field of interest

2. Peeling - self-cal. on confusing source (to remove phase errors), get model & subtract source. Return to original calibration & insert model into visibilities

3. Direction-dependent calibration - see Advanced Imaging lecture

These are listed in order of complexity - note that direction dependent calibration is not available for all telescope arrays

CONFUSION

1. Outlier fields

If the source is out of your desired target area, then you can set a small area around the confusing source and deconvolve with the main image.

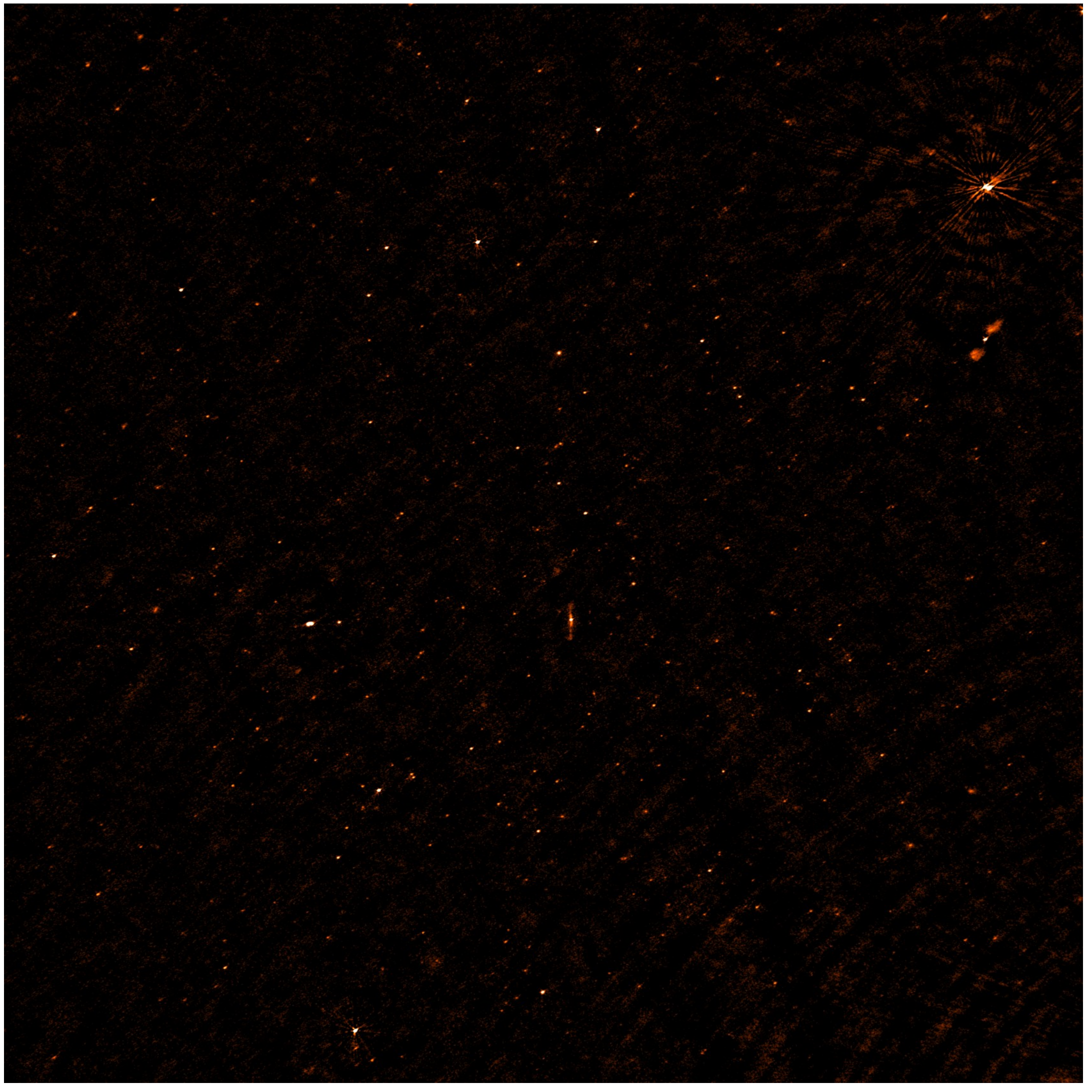
In CASA, this is achieved by setting multiple images (see right) or set an outlier file (orange box & example below)

```
#  clean :: Invert and deconvolve images with selected algorithm
vis                 = 'JVLA_combined_GOODSN.ms' #  Name of input visibility file
imagename           = ['main', 'outlier'] #  Pre-name of output images
outlierfile         =         ''           #  Text file with image names, sizes, centers for
                                           #   outliers
field               =         ''           #  Field Name or id
spw                 =         ''           #  Spectral windows e.g. '0~3', '' is all
selectdata          =       True           #  Other data selection parameters
     timerange      =         ''           #  Range of time to select from data
     uvrange        =         ''           #  Select data within uvrange
     antenna        =         ''           #  Select data based on antenna/baseline
     scan           =         ''           #  Scan number range
     observation    =         ''           #  Observation ID range
     intent         =         ''           #  Scan Intent(s)

mode                =      'mfs'           #  Spectral gridding type (mfs, channel, velocity,
                                           #   frequency)
     nterms         =          1           #  Number of Taylor coefficients to model the sky
                                           #   frequency dependence
     reffreq        =         ''           #  Reference frequency (nterms > 1),'' uses central
                                           #   data-frequency

gridmode            =         ''           #  Gridding kernel for FFT-based transforms, default=''
                                           #   None
niter               =        500           #  Maximum number of iterations
gain                =        0.1           #  Loop gain for cleaning
threshold           =   '0.0mJy'           #  Flux level to stop cleaning, must include units:
                                           #   '1.0mJy'
psfmode             =    'clark'           #  Method of PSF calculation to use during minor cycles
imagermode          =  'csclean'           #  Options: 'csclean' or 'mosaic', '', uses psfmode
     cyclefactor    =        1.5           #  Controls how often major cycles are done. (e.g. 5
                                           #   for frequently)
     cyclespeedup   =         -1           #  Cycle threshold doubles in this number of iterations

multiscale          =         []           #  Deconvolution scales (pixels); [] = standard clean
interactive         =      False           #  Use interactive clean (with GUI viewer)
mask                =         []           #  Cleanbox(es), mask image(s), region(s), or a level
imsize              = [[8000, 8000], [50, 50]] #  x and y image size in pixels.  Single value:
                                           #   same for both
cell                = ['0.33arcsec']       #  x and y cell size(s). Default unit arcsec.
phasecenter         = ['J2000 12h36m49.4 62d12m58.0', 'J2000 12h34m52.2 62d02m34.53'] #  Image
```

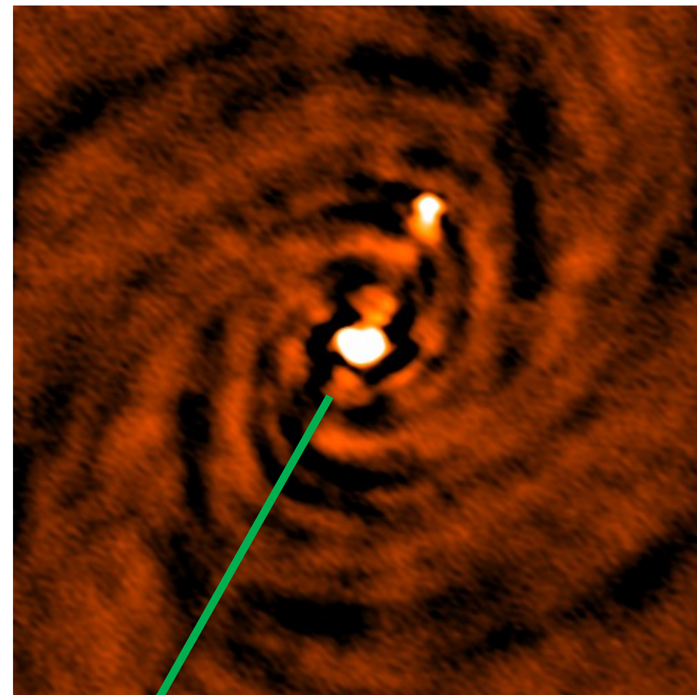
```
#content of outliers.txt
#
#outlier field1
imagename='outlier1'
imsize=[512,512]
phasecenter = 'J2000 12h34m52.2 62d02m34.53'
mask='box[[245pix,245pix],[265pix,265pix]]'
```

CONFUSION

1. Outlier fields



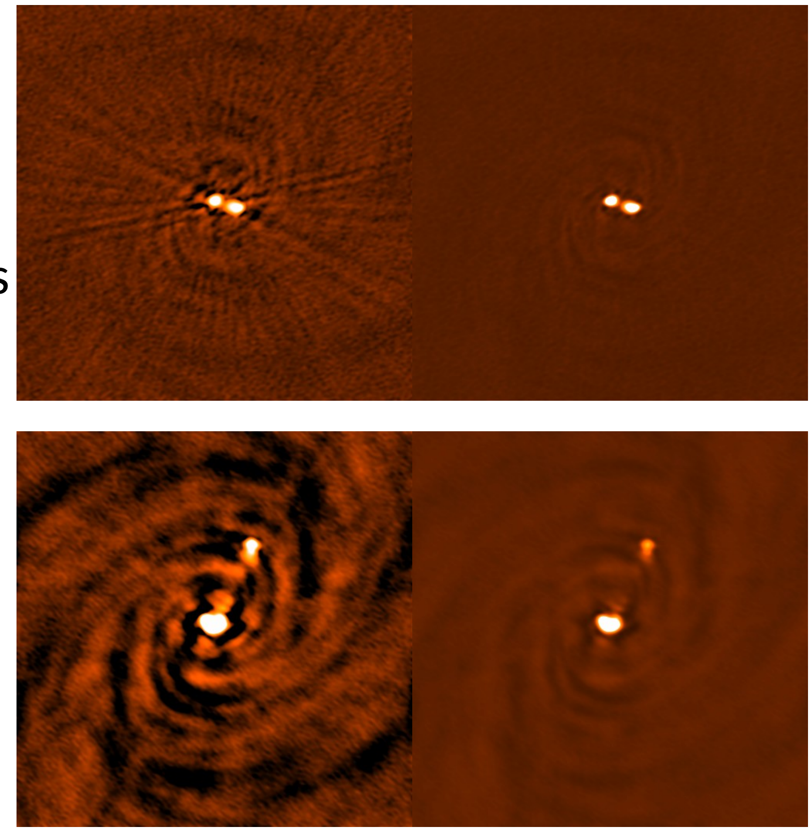
0.25 Jy confusing source using outlier field assigned



CONFUSION

2. Peeling If outlier fields do not work try peeling!

- After phase calibrating the data, perform self-calibration for the brightest confusing source – then subtract it out
- Delete phase solutions derived for previous confusing source (1)
- Move to next brightest confusing source, perform self-calibration/imaging cycles – then subtract that source from the dataset (2)
- Perform (1) and (2) until all confusing sources are subtracted. Delete all self-calibration solutions and image central regions

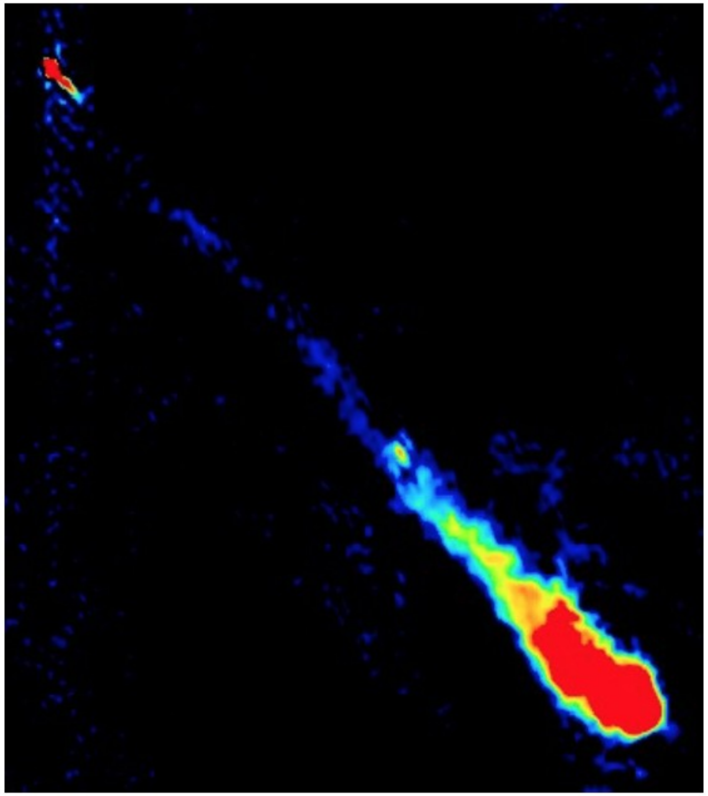


Before After

HIGH DYNAMIC RANGE IMAGING

ASTRON

- Present dynamic range limits (on axis):
  - Phase calibration – up to 1000:1 ! improve with self-calibration
  - Non-closing data errors – continuum ~20,000:1, line >100,000:1
  - After non-closing error correction ~10,000,000:1
- Non-closing errors thought to be dominated by small changes in telescope passbands.
- Spectral line data configurations are the default for all new wide-band radio telescopes.
- In order to subtract out confusion we will need to be able to image with these very high dynamic ranges away from the beam centre



3C273, Davis et al. (MERLIN)
1,000,000:1 peak – RMS

Credit T. Muxlow

SIGNAL TO NOISE

Noise level of a (perfect) homogeneous interferometer:

$$\text{Noise} = \frac{\sqrt{2}k_B T_{\text{sys}}}{\sqrt{n_b t \Delta\nu} A\eta}$$

where:
- T_{sys} - system temperature [K]
- n_b - number of baselines
- t - integration time [s]
- $\Delta\nu$ - bandwidth [Hz]
- A - area of apertures [m]
- η - aperture efficiency

Many factors increase noise level above this value:

- Confusion
- Calibration errors
- Bad data
- Non-closing data errors
- Deconvolution artefacts

Rarely get this from an image. Dependent of flagging accuracy, calibration & adequate deconvolution

But techniques presented in this workshop can get you closer!

CONCLUSION

- interferometry samples Fourier components of sky brightness
- make an image by Fourier transforming sampled visibilities
- deconvolution attempts to correct for incomplete sampling

- remember
  - there are an infinite number of images compatible with the visibilities
  - missing (or corrrupted) visibilities affect the entire image
- astronomers must use judgement in the imaging and deconvolution process
- it's fun and worth the trouble → high angular resolution images!

many, many issues not covered in this talk: see References and upcoming talks